

ProjectStructure

Apache JDO Project Structure

Apache JDO uses maven for project builds. Each sub-project appears as a directory underneath the top level jdo directory. Each sub-project contains some combination of:

- a src directory which contains the Java sources
- a test directory which contains the Java sources for the JUnit tests
- an xdocs directory which contains the maven site building documents
- project.properties maven file
- project.xml maven file
- maven.xml maven file

Dependencies

- Apache JDO uses maven for builds. Get more information about maven [here](#).
- Apache JDO uses maven and defines maven artifacts. See [MavenInfo](#).

Sub-Projects

The following sub-projects have been defined:

directory	groupid	artifactid	description
api11	javax.jdo	jdo1-api	JDO 1.0 API. This is the standard definition of the JDO API as defined by the JSR-12 standard. No additional work is expected in this project.
ri11	org.apache.jdo	jdo1-ri	JDO 1.1 Reference Implementation. This is a file-based, single-user store for persistent Java objects. The storage component is called File Object Store (FOStore). The version number is 1.1 to reflect that the Java package has changed from com.sun.jdori to org.apache.jdo. The license has also changed.
btree	org.apache.jdo	jdo-btree	This sub-project contains an adapter to the btree open source implementation developed by Netbeans. The maven build for this directory downloads the btree package sources from Netbeans and builds it. The ri11 project depends on this sub-project.
tck11	org.apache.jdo	jdo1-tck	JDO 1.1 Technology Compatibility Kit. This set of programs tests that a JDO 1.0 implementation is in compliance with the standard. The version number is 1.1 to reflect that the test framework has changed from JavaTest (a proprietary Sun package not suitable for open source projects) to JUnit , an open source test framework. The license has also changed.
api20	javax.jdo	jdo2-api	JDO 2.0 API. This is the standard definition of the JDO API as defined by the JSR-243 standard.
tck20	org.apache.jdo	jdo2-tck	JDO 2.0 Technology Compatibility Kit. This set of programs tests that a JDO 2.0 implementation is in compliance with the standard.
fore20	org.apache.jdo	jdo2-fstore	JDO 2.0 FOStore (File Object Store) implementation. This is an implementation of JDO 2.0 based on the JDO 1.0 Reference Implementation. It does not support relational data access, so it is inappropriate for the JDO 2.0 Reference Implementation. See below.
core20	org.apache.jdo	jdo2-core	JDO 2.0 core components implementation. This is the core of an implementation of JDO 2.0 based on the JDO 1.0 Reference Implementation. It includes the model of persistence and other utility functionality.
query20	org.apache.jdo	jdo2-query	JDO 2.0 query implementation. This is the query component responsible for parsing JDOQL and generating the query plan and execution.
enhancer20	org.apache.jdo	jdo2-enhancer	JDO 2.0 enhancer implementation. This is the enhancer component of an implementation of JDO 2.0 based on the JDO 1.0 Reference Implementation. It is used to modify class files so they can be used in a binary-compatible implementation of JDO 2.0.
geronimo	org.apache.jdo	jdo2-geronimo	JDO 2.0 Geronimo Integration. This project will build the glue between the JDO 2.0 Reference Implementation and the J2EE compliant application server. We expect to use a common persistence framework (Tranql) to provide both EJB3 and JDO persistence services. We also expect to allow FOStore to be able to be defined as a Connector, and looked up as a JNDI resource.